IN THE TITLE:

Above the Title, please add:

<u>TITLE OF THE INVENTION</u>

IN THE SPECIFICATION:

On page 1 of the Specification, please delete:

~~LITERATURE~~

~~Patents cited in [opposition]~~

~~U.S. Patent:~~

~~5,694,542 ———— December 2, 1997 ———— Kopetz, H.~~

~~European Patent:~~

~~0658 257 ———— December 18, 1996 ———— Kopetz, H.~~

~~Kopetz, H. (1997). *Real-Time Systems, Design Principles for Distributed Embedded*~~

~~*Applications*; ISBN: 0-7923-9894-7. Boston. Kluwer Academic publishers.~~

~~Maekawa, M. (1987) et. al., *Operating Systems*, ISBN 0-8053-7121-4, Menlo Park, Cal,~~

~~Benjamin Cummings Publishing Company.~~

~~CAN (1990). *Controller Area Network CAN, an In-Vehicle Serial Communication Protocol.*~~

~~SAE Handbook 1992, SAE Press. p. 20.341-20.355, Society of Automotive Engineers,~~

~~Warrendale, PA USA.~~

~~IOOP (1994). *OMG's Internet Inter-ORB Protocol (IIOP)*, Object Management Group,~~

~~Internet: www.omg.org, Boston, USA~~

Please insert therefore:

CROSS-REFERENCE TO RELATED APPLICATION

This application claims priority from Austrian Patent Application A 429/2001, filed 19 March 2001, and PCT Patent Application Serial No. AT02/00090, filed 19 March 2002.

On page 2 of the Specification, please delete

~~ABSTRACT~~

and insert therefore:

BRIEF SUMMARY OF THE INVENTION

On page 4 of the Specification, please delete

~~DESCRIPTION OF AN EMBODIMENT~~

And insert therefore:

DETAILED DESCRIPTION OF THE INVENTION

Please amend the second paragraph as shown:

Status data are data that provide information about the observed value of status variables. An observation of a status variable is an indivisible triple

<Name of the status variable, Value of the status variable, Moment of the observation>

as in described in detail in *Real-Time Systems, Design Principles for Distributed Embedded Applications*; ISBN: 0-7923-9894-7. Boston. Kluwer Academic publishers. Kopetz 1997, p. 31. An example of a status element is the present position of a valve. The semantics of the status data suggest a new value of a status variable overwrites the existing old value and the value is not consumed as it is read, that is, the same value can be read multiple times. The possibility presents itself of configuring the interface between two subsystems that communicate via status data as a (dual-ported) memory interface. The transmitter must make

3

sure that the currently valid value of a status variable is available in a data storage memory on the receiver side. A new status data value can overwrite the existing old value. The receiver, if it ever needs the value, can read out the current value from the local memory without consuming it via an *information pull* command.

On pages 7 and 8 of the Specification, please amend the paragraph as follows:

Subsystem 310 is made up of a ring buffer store, in the particular case with eight memory locations, the four memory locations 312 on the left being open in Fig. 3 and the four memory locations 313 on the right being assigned data elements. Connected to the ring buffer store are the two pointers 311 and 314. Pointer 311 points to the next data element to be consumed. After the consumption of a data element by the host computer, the pointer is changed so that it points toward the next data element to be consumed. Pointer 314 points to the next open memory location. After a new data element is saved by the communication system at the transmitter or receiver, pointer 314 is changed so that it points to the next open memory location. The detailed design of a ring buffer store is prior art and is described in detail in standard text books about operating systems, for example, <u>*Operating Systems*, ISBN 0-8053-7121-4, Menlo Park, Cal, Benjamin Cummings Publishing Company. (1987)</u> Maekawa et. al., p. 21. The desired event semantics are implemented via the ring buffer store logic. In order to prevent a loss of data, the ring buffer store size must be matched to the processing frequency of the host computer in such a manner that an open place is always available for a newly arriving event message. If this is not the case, a fault situation is imminent (if the two pointers 311 and 314 run in conjunction) and is to be signaled to the host computer via an interrupt signal. In a correctly dimensioned system with an appropriate ring buffer store, even in a system that transmits event data to the host computer, there is no unplanned interruption. This is important, because any unplanned interruption violates the

4

assumptions on which the WCET (worst case execution time analysis) of the host computer is

based. The previously described method for the processing of event messages implements the

required *exactly-once* semantics of the event data. In special cases, the ring buffer store can

also have length 1.